

## 百度 2014 校园招聘笔试题 ——深度学习算法研发工程师.

今天上午到腾讯那里,没有叫到笔试通知,是去霸笔的。原本以为是可以霸笔的,因为像阿里,360 等大公司都专门的考场给霸笔的人准备的,但是腾讯不是这样,只有两个很小教室给那些霸笔的人考试,一会就都占满了,太少了,还有一大批没收到通知人不让考试。参加腾讯笔试的也够火爆的,一条走廊全占满了人,都是霸笔的,等了半小时者不让进,太不厚道了。

下午到百度的笔试,比腾讯大气多了,给那些霸笔的人提供了 3 个大教室,每个教室能容纳 100 多人,而且有专门的人安排座位。百度的职位真多啊,有一二十个,我选择的是 B13 深度学习算法研发工程师,一个教室还就我一个人选这个职位。试卷一发下来,我愣了,真难度也太大了,好多我见都没见过……

现将题目贡献如下:

### 一、简答题

1. 深度神经网络目前有哪些成功的应用? 简述原因。(10 分)
2. 列举不同进程共享数据的方式(至少三种)。(10 分)
3. 对于  $N$  个样本,每个样本为  $D$  维向量,采用欧式距离使用 KNN 做类预测。(10 分)
  - 1). 给出预测时间复杂度。
  - 2). 当  $N$  很大时,有哪些方法可以降低复杂度?
  - 3).  $k$  取值的大小对预测方差和偏差有何影响?

## 二、算法和程序设计

1. 给出一个数据  $A=[a_0, a_1, a_2, \dots, a_n]$ (其中  $n$  可变), 打印出该数值元素的所有组合。(15 分)

2. 有这样一个数组  $A$ , 大小为  $n$ , 相邻元素差的绝对值都是 1, 如  $A=\{4,5,6,5,6,7,8,9,10,9\}$ 。现在给定数组  $A$  和目标整数  $t$ , 请找到  $t$  在数组中的位置。(15 分)

3. 在平面上有一组间距为  $d$  的平行线, 将一根长度为  $l(l < d)$  的针任意掷在这个平面上, 求此针与平行线中任意一根相交的概率, 用高等数学(微积分、概率的方法)求解, 基于布丰投针的结论, 任选一种编程语言(C/C++, matlab, python, java), 写出模拟投针实验(程序中允许把一个理想的  $\pi$  作为常量使用), 求解圆周率。(15 分)

## 三、系统设计题(两题中任选一题作答, 25 分)

1. 假设有数据集  $(x, t) = \{(x_1, t_1), (x_2, t_2), (x_3, t_3), \dots, (x_i, t_i)\}$  和如下的神经网络模型

$$h^{[i]} = \sigma(w^{[i]}h^{[i-1]} + b^{[i]}), i = 1, 2, 3$$

其中  $\sigma(x) = \max(x, 0)$

$w^{[i]}$  为该神经网络的第  $i$  层权重矩阵,  $b^{[i]}$  为第  $i$  层偏移量,  $h^{[i]}$  为第  $i$  层的输出

$$y_{ij} = \frac{e^{h_{ij}^{[3]}}}{\sum_{j=1}^N e^{h_{ij}^{[3]}}}$$

损失函数为

$$L = \sum_{i=1}^l \sum_{j=1}^N t_{ij} \log(y_{ij}) + c \sum_{i=1}^3 \|w^{[i]}\|^2$$

求  $\frac{\partial L}{\partial w^{[i]}}$ ,  $\frac{\partial L}{\partial b^{[i]}}$ ,  $i=1, 2, 3$

2. 关于 K-means 聚类算法, 请回答以下问题:

1). 写出将  $N$  个样本  $X=(x_1, \dots, x_N)$  聚类成  $k$  类的  $k\_means$  聚类算法的优化目标;

2). 描述 K-means 终止的常用条件;

更多企业校园招聘笔试面试题合集下载: <http://bimian.xuanjianghui.com.cn/>

3). 以 Kmeans 算法为例, 描述 Expectation-Maximization(EM)算法的基本原理与步骤。

4). 用伪代码给出基于 MPI 或者 HADOOP 的 Kmeans 并行算法。

## 我的部分解答:

二、(2)

[java] view plaincopyprint?

```
1. public static int find(int[] a, int t) {
2.     int i=0;
3.     while(i<a.length) {
4.         if(t == a[i]) {
5.             return i;
6.         } else {
7.             i += Math.abs(t - a[i]);
8.         }
9.     }
10.    return -1;
11. }
12.
13. public static void main(String[] args) {
14.     int[] a = {4,5,6,5,6,7,8,9,10,9};
15.     System.out.println(find(a, 7));
16. }
```